Identification of Rare & Novel Senses Using Translations in a Parallel Corpus

Richard Schwarz

Institute for Natural Language Processing (IfNLP)
University of Stuttgart

LREC 2010

Intuition

Observation

Different senses of polysemous words have different translations.

Example

sentence $\mapsto$ *peine* (law) or *phrase* (ling)

Intuition

Observation

Different senses of polysemous words have different translations.

Example

sentence $\mapsto$ *peine* (law) or *phrase* (ling)

Claim

Common uses will mostly be translated with the same words, i.e. the set of translations will be homogeneous. For rare & unusual uses the variance is high and therefore the set of translations will be heterogeneous.

Basic Approach

Task

Find rare & novel uses of French verbs.

1. take parallel corpus with many languages (Europarl)
2. align sentences and words (Giza++)
3. for each verb, cluster its senses according to its translation equivalents
4. inspect heterogeneous clusters to find rare uses

Vector Representation

Idea

Treat each occurrence of a verb as a vector, where the dimensions are the translations.

Example

Occurrence no. 453 of demander

demander

Translations

ask (English)
fragen (German)
chiedere (Italian)
preguntar (Spanish)
vraag (Dutch)
perguntar (Portuguese)
spørge (Danish)
fråga (Swedish)

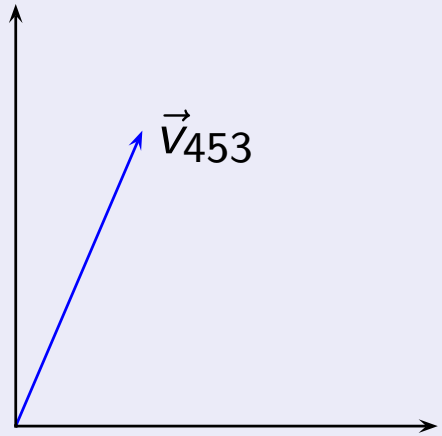
Vector Representation

$$\vec{v}_{453}(\text{demander}) = \begin{pmatrix} \text{ask} \\ \text{fragen} \\ \text{chiedere} \\ \text{preguntar} \\ \text{vraag} \\ \dots \end{pmatrix}$$

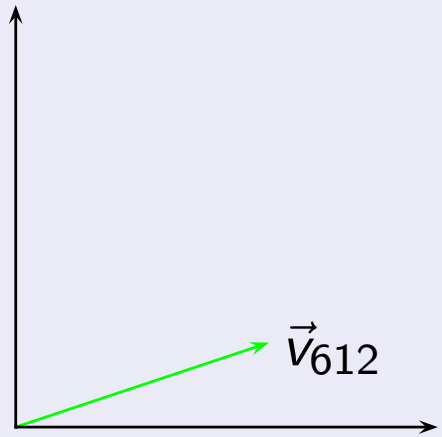
$$\vec{v}_{612}(\text{demander}) = \begin{pmatrix} \text{urge} \\ \text{fordern} \\ \text{invito} \\ \text{insto} \\ \text{dring} \\ \dots \end{pmatrix}$$

Vector Representation

$$\vec{v}_{453}(\text{demander}) = \begin{pmatrix} \text{ask} \\ \text{fragen} \\ \text{chiedere} \\ \text{preguntar} \\ \text{vraag} \\ \dots \end{pmatrix}$$



$$\vec{v}_{612}(\text{demander}) = \begin{pmatrix} \text{urge} \\ \text{fordern} \\ \text{invito} \\ \text{insto} \\ \text{dring} \\ \dots \end{pmatrix}$$



Term Weighting

Observation

Translations differ in importance for sense discrimination.

Term Weighting

Observation

Translations differ in importance for sense discrimination.

Example

For the verb *demander*, the alignments *the*, *ask*, *survey* and *whether* clearly have different amounts of information for our task.

Term Weighting

Observation

Translations differ in importance for sense discrimination.

Example

For the verb *demander*, the alignments *the*, *ask*, *survey* and *whether* clearly have different amounts of information for our task.

Definition

weight$(t|v) = ?$

Term Weighting

Definition

af(v, t) = alignment frequency
number of times translation t and verb v are aligned

- af(demander, ask) $>$ af(demander, whether)
- high af suggests valid/significant translation

Term Weighting

Definition

af(v, t) = alignment frequency
number of times translation t and verb v are aligned

- af(demander, ask) $>$ af(demander, whether)
- high af suggests valid/significant translation
- but what if af(demander, the) $>$ af(demander, survey)

Term Weighting

Definition

af(v, t) = alignment frequency
number of times translation t and verb v are aligned

- af(demander, ask) > af(demander, whether)
- high af suggests valid/significant translation
- but what if af(demander, the) > af(demander, survey)

Definition

cf(t) = corpus frequency
number of times term t occurs in corpus

- cf(the) > cf(survey)
- high cf weakens validity of af

Term Weighting

Definition

$$\text{weight}(t|v) = \max\left(\frac{\text{af}}{n}, \frac{\text{af}}{\text{cf}}\right) \quad (n = \text{number of vectors})$$

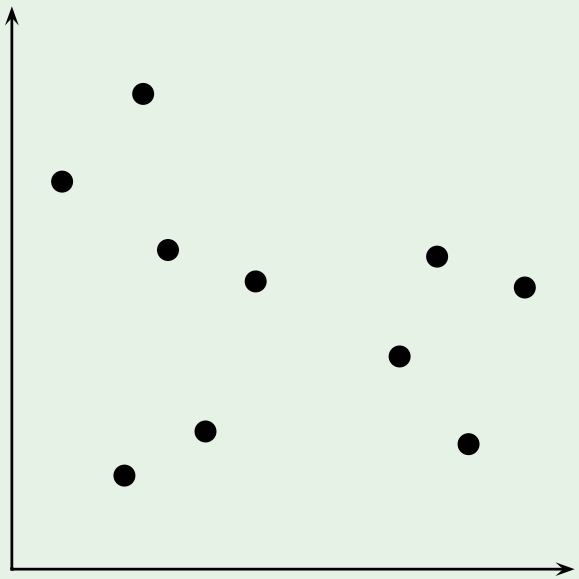
Definition

$$\text{sim}(a, b) = \frac{a \cdot b}{||a||\,||b||} \quad \text{(cosine similarity)}$$

Graphical Representation of Vector Space

Example

The figure shows an example vector space with $n = 10$.



Cluster

Description of a Cluster

- collection of objects that are similar to each other
- has a centroid (arithmetic mean over all objects in the cluster)

Approach

1. group similar vectors into small clusters with maximum size m, using custom N-Secting K-means
2. group resulting clusters into K bigger superclusters, using standard K-Means (on centroids of clusters)

N-secting K-means

Idea

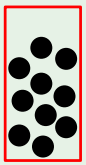
Group similar vectors into small clusters, with the condition that no cluster exceeds a pre-defined size limit m.

Example

initial state: all vectors in one cluster

$n = 10$, number of vectors

$m = 3$, maximum number of vectors per cluster

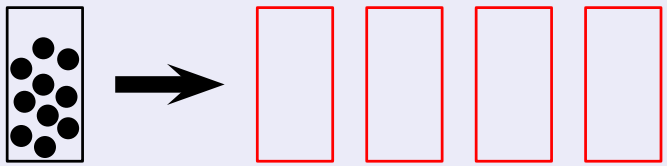


N-secting K-means

Step 1

N = smallest number that can satisfy the condition $= \lceil \frac{n}{m} \rceil = \lceil \frac{10}{3} \rceil = 4$

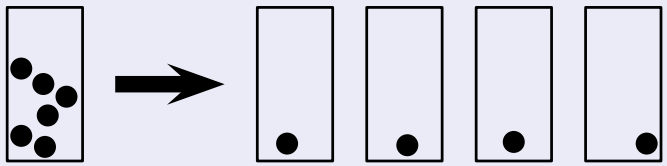
create N new clusters



N-secting K-means

Step 2

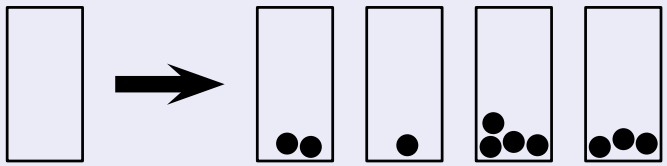
randomized initial assignments



N-secting K-means

Step 2

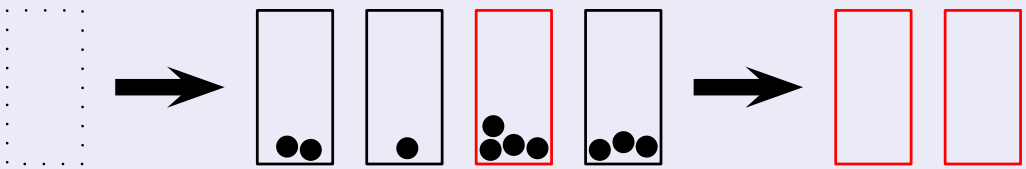
assign vectors according to centroid similarity



N-secting K-means

Step 3

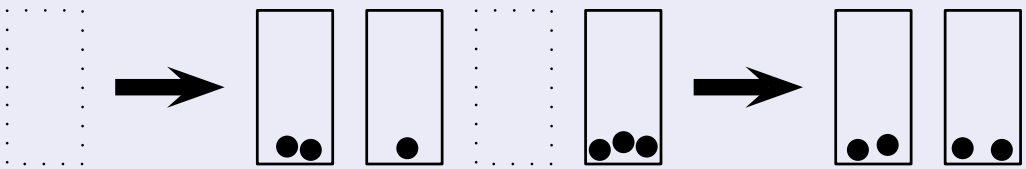
repeat until condition is no longer violated



N-secting K-means

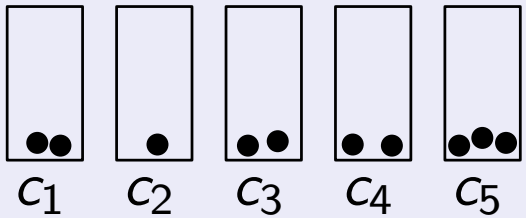
Step 3

repeat until condition is no longer violated



N-secting K-means

Final result



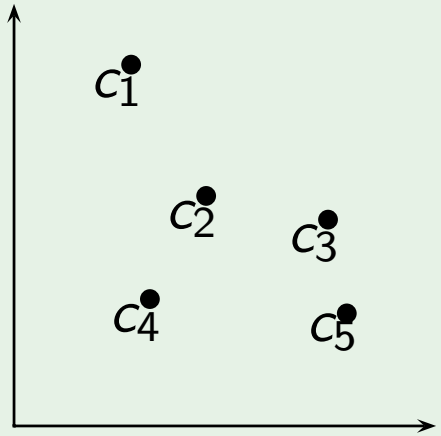
K-means

Idea

Group similar clusters into K superclusters, by applying standard K-means on cluster centroids.

Example

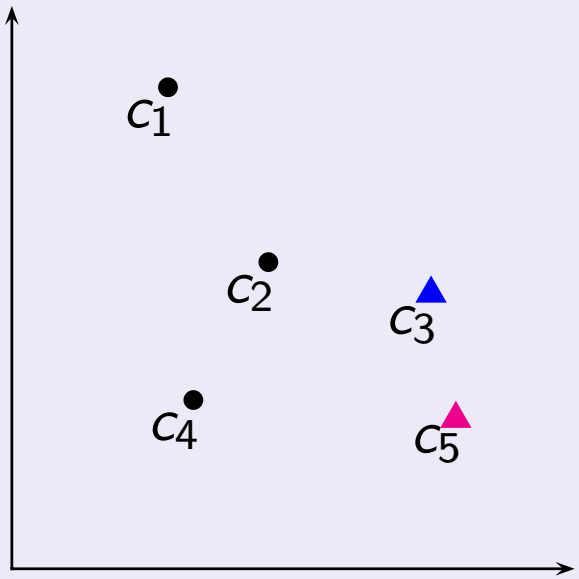
$K = 2$, number of resulting superclusters



K-means

Step 1

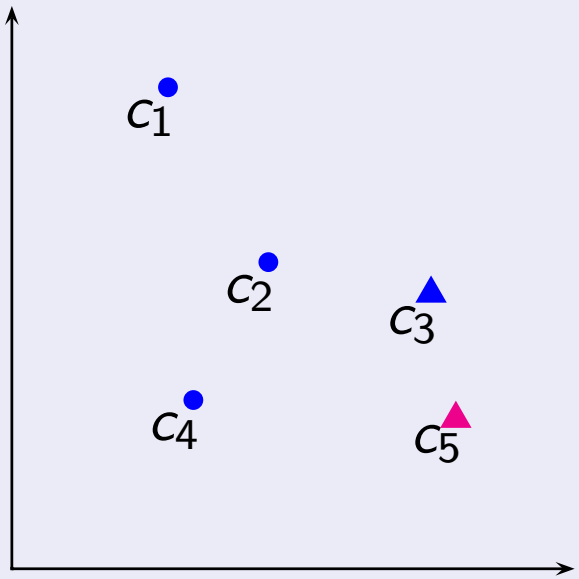
randomly select K initial centroids



K-means

Step 2

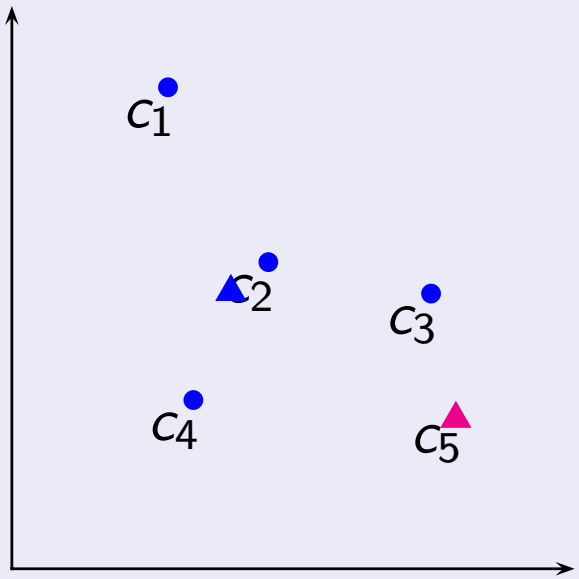
assign clusters to supercluster with nearest centroid



K-means

Step 3

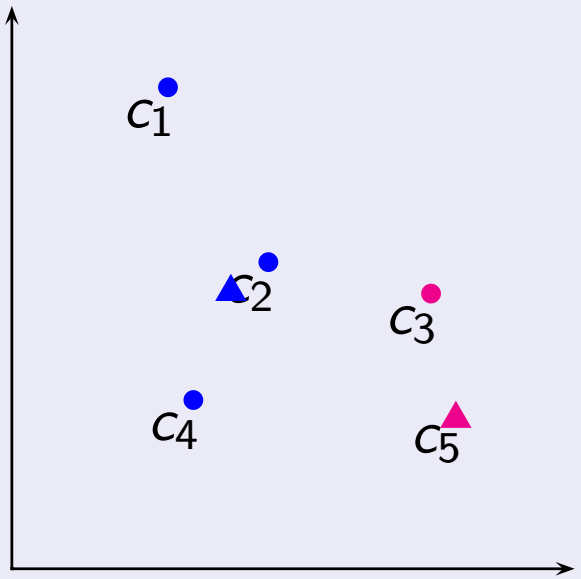
re-compute centroids



K-means

Step 4

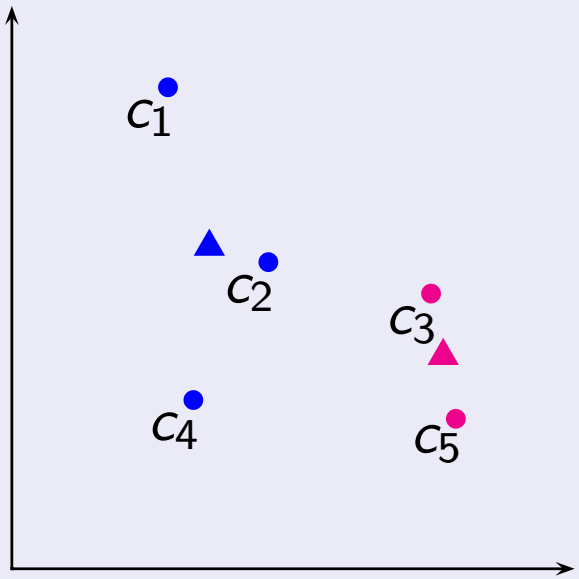
re-assign clusters to nearest supercluster



K-means

Step 5

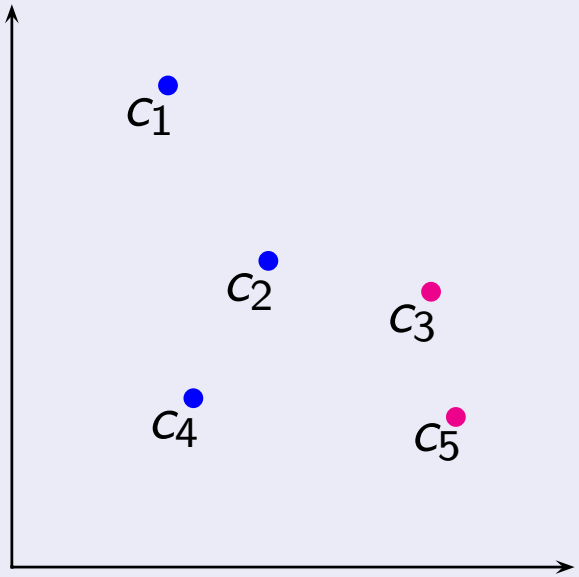
re-compute centroids



Result

Final result

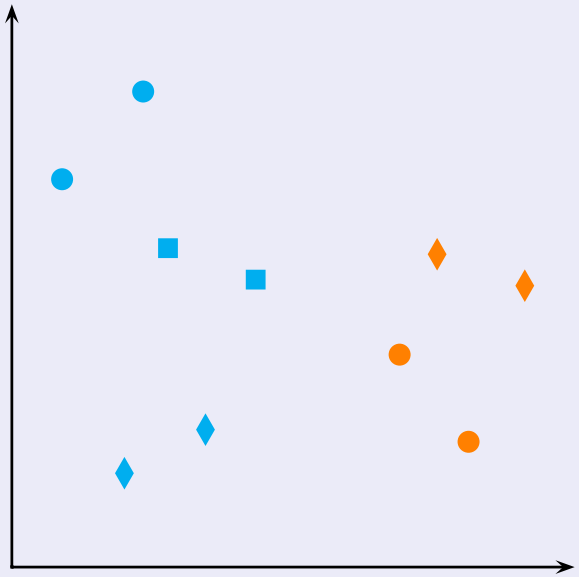
in terms of clusters



Result

Final result

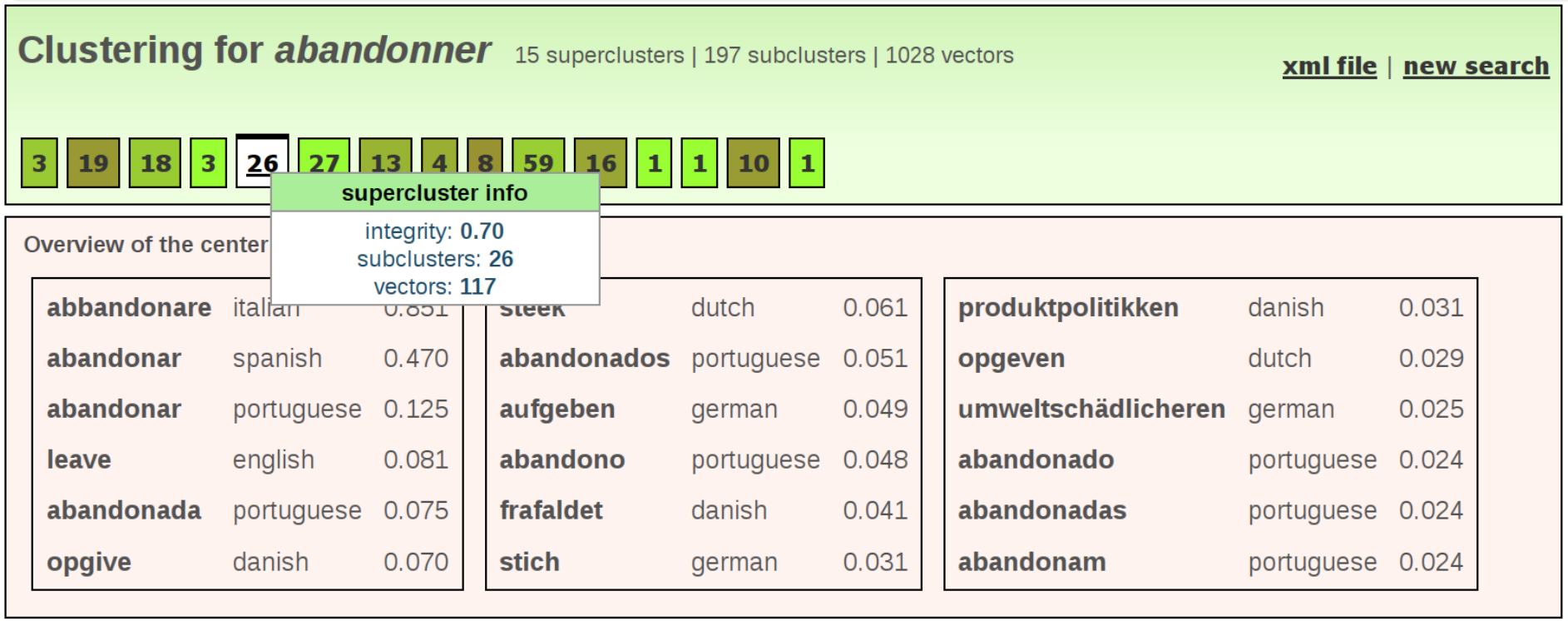
in terms of vectors



Glance at the User Interface

Webinterface

The figure demonstrates the view of a supercluster for the query "abandonner". The lexicographer can browse through the different superclusters by clicking the boxes.

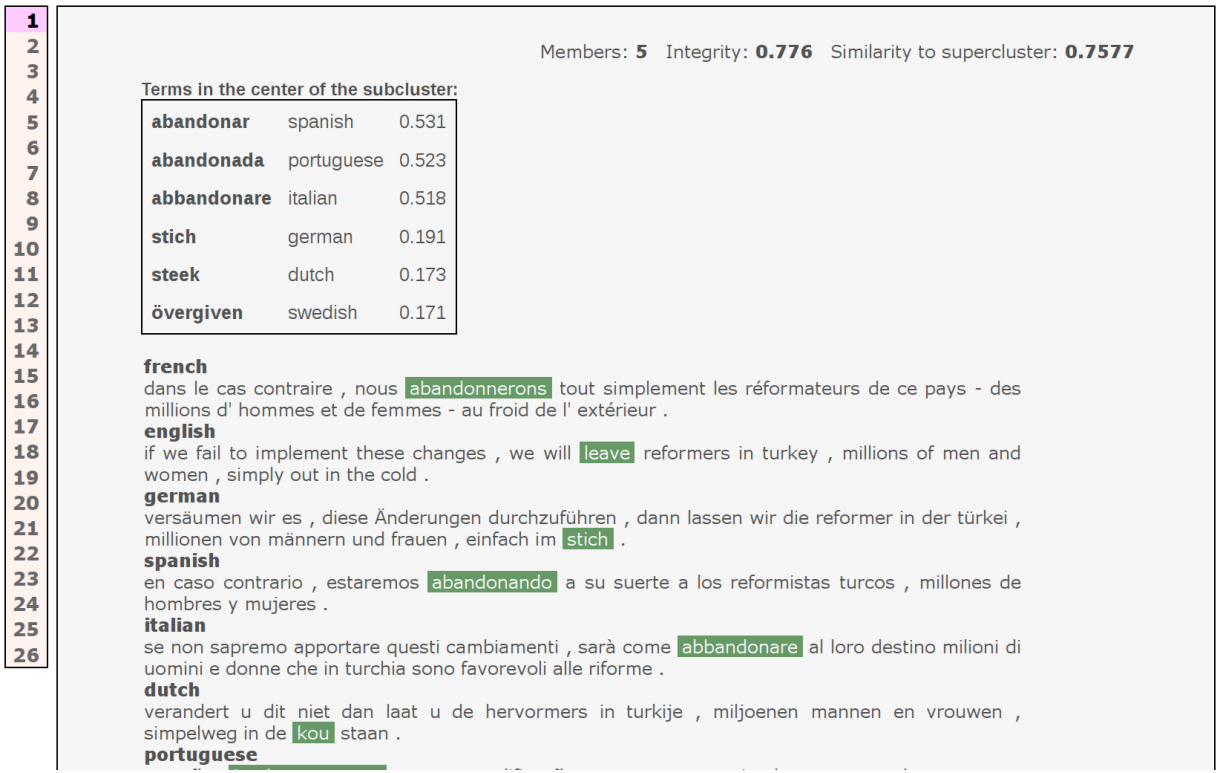


- homogeneity is color coded, allowing for a fast overview

Glance at the User Interface

Webinterface

The figure shows the view of a cluster. Giving a basic overview of the centroid and listing all sentences. Translations of the query are highlighted.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

Members: **5** Integrity: **0.776** Similarity to supercluster: **0.7577**

Terms in the center of the subcluster:

abandonar	spanish	0.531
abandonada	portuguese	0.523
abbandonare	italian	0.518
stich	german	0.191
steek	dutch	0.173
övergiven	swedish	0.171

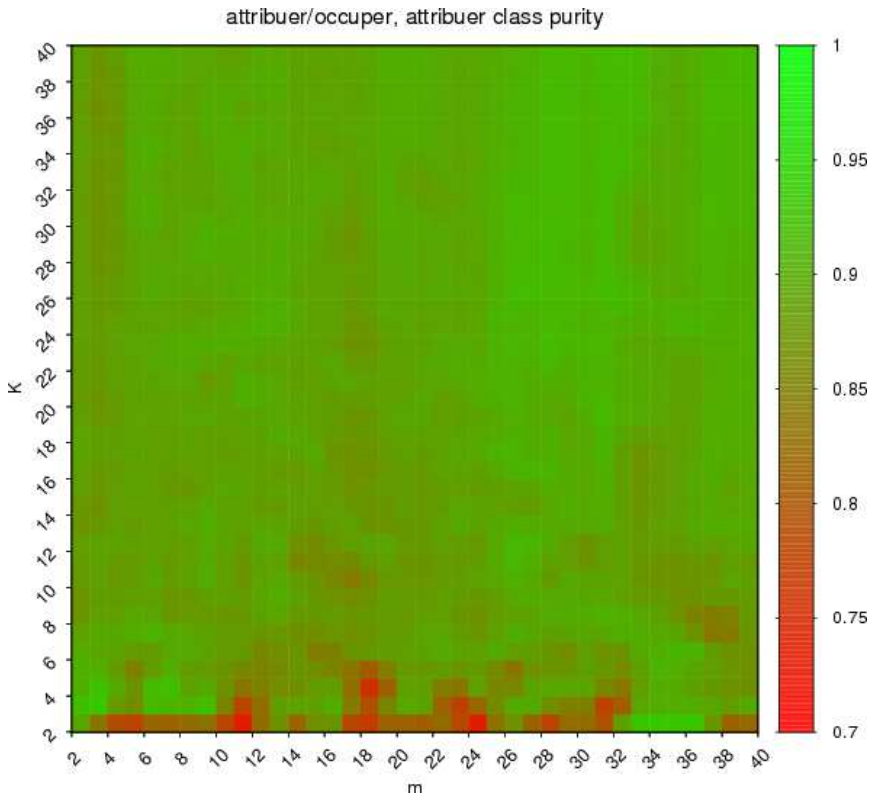
french
dans le cas contraire , nous abandonnerons tout simplement les réformateurs de ce pays - des millions d' hommes et de femmes - au froid de l' extérieur .
english
if we fail to implement these changes , we will leave reformers in turkey , millions of men and women , simply out in the cold .
german
versäumen wir es , diese Änderungen durchzuführen , dann lassen wir die reformer in der türkei , millionen von männern und frauen , einfach im stich .
spanish
en caso contrario , estaremos abandonando a su suerte a los reformistas turcos , millones de hombres y mujeres .
italian
se non sapremo apportare questi cambiamenti , sarà come abbandonare al loro destino milioni di uomini e donne che in turchia sono favorevoli alle riforme .
dutch
verandert u dit niet dan laat u de hervormers in turkije , miljoenen mannen en vrouwen , simpelweg in de kou staan .
portuguese

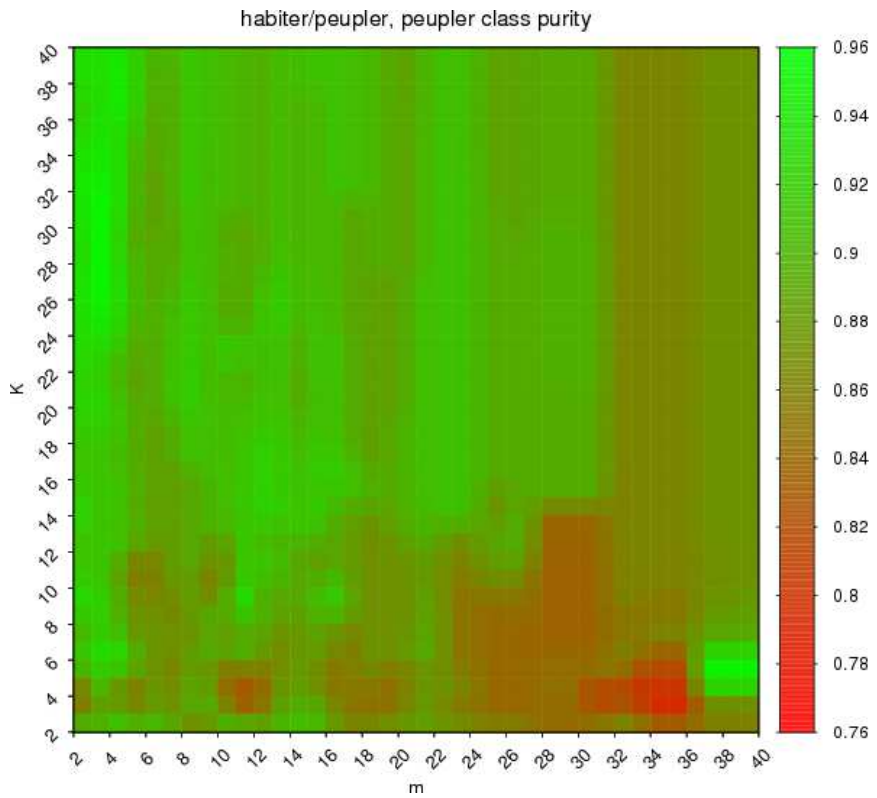
Evaluation Using Pseudowords

Idea

Use union of vectors of two distinct queries. Calculate purity of resulting superclusters, to show that the clustering itself is reasonable.

- did this for several verb pairs with always high purity





Qualitative Evaluation

Idea

Manually go through noisy superclusters to find rare or novel uses.

Findings

did this for *abandonner*, *glisser*, *mobiliser*, *parcourir*, *payer*, *remercier*

- found new uses, including:
  - abandonner qc en faveur de qc
  - glisser qc à qn (in the sense "give")
  - faire payer (in the sense "charge")
  - payer qc sur qc (in the sense "pay something on something else")

Conclusions

Problems

- word alignment difficult, especially for verbs and long sentences
- domain restriction of corpus limits possible senses (e.g., no sexual or funny senses)
- clustering algorithms find local (not global) optimum

Conclusions

- very distinct senses are successfully discriminated
- frequent and regular uses grouped into homogeneous clusters
- rare uses are segregated from regular uses
- $\Rightarrow$ lexicographer can discard most occurrences quickly
- can be easily applied to other languages